How to work in the seminar?

1. **Choose a topic** (in this first session), see the seminar homepage
2. **Search additional literature**, as needed
3. Write a **10-15 page long overview** of the topic (use the department’s template) and send it in good time to the participants (a week before!)
4. Make a **presentation** in class based on your overview
5. **Comment** on other’s writings and be **active** in the presentations
   - To pass the seminar, you need to attend **at least 3/4** of the sessions
   - Your **grade** depends on the written text, the presentation and session activity (commenting the writings and presentations)
   - The language of the seminar is English, so text and presentation are in English (use a spell checker!)
Timetable

• The seminar convenes on Mondays 14-16 in C222

Short introduction

• I will now make a short introduction into the topic
• We will then proceed to dividing the topics between the participants
The recommendation problem

- Given a set $C$ of users, a set $S$ of possible items, and a utility function $u: C \times S \rightarrow R$ with $R$ being an ordered set, we want for each user to choose item $s'$ so that $u(c,s')$ is maximised [or to get the $N$ best]
- $S$ can be films, books, grocery items etc.
- $u$ can be explicit information e.g. user ratings or implicit information e.g. decisions to buy a product
- Note that recommendations are personalised by user $c$. Just listing the most popular items is not personalised

Characteristics of the problem

- Typically the set of items $S$ is very large (even millions of items); also the set of users $C$ can be very large $\Rightarrow$ scalability is a real problem
- Each $c$ can be defined with a profile: age, gender, etc.; similarly $s$ can be defined with a set of characteristics
- The input matrix $C \times S$ is usually very sparse: $u$ is given only on a small subset
- The problem is actually to extrapolate to missing values: fill in the matrix $C \times S$ and choose the best
Basic approaches (1/2)

- **Collaborative** recommendations/filtering: recommend items that users with similar tastes preferred in the past
  - “user-based” if we look for similar users; “item-based” if we look for similar items
- **Content-based** recommendations/filtering: recommend items similar to the ones the user preferred in the past (looking at characteristics of the items, like genre, author etc.)

Basic approaches (2/2)

- **Knowledge-based** recommendations: recommend items based on the specific domain knowledge about how certain item features meet user needs and preferences
- **Hybrid** approaches: Combine methods
- (More approaches exist)
Collaborative recommendations (1/3)

• How to measure similarity?
  – Many different possibilities, e.g.
    • Pearson’s correlation coefficient for similarity between users
    • Adjusted cosine similarity for similarity between items \( a \) and \( b \): take cosine of the angle between the two vectors of “ratings minus average ratings” for \( a \) and \( b \)
  
• To speed up calculations, only the closest neighbourhood is considered

• Similarity calculation ⇒ neighbourhood selection ⇒ selecting the best

Collaborative recommendations (2/3)

• Problems with CF:
  – What to do with new users with no history (“cold start”)?
    • We could use demographic data
  – What to do with new items not rated yet?
  – Maybe misleading e.g. buying a book for another person
  – Opinions on controversial items may be more interesting
  – Sparseness: almost no ratings
Collaborative recommendations (3/3)

- **Memory-based** (use data directly) and **model-based** techniques (make offline a model of data reducing the data size and use that for the calculations)
- We can view model-based methods as a normal classification problems using methods from machine learning: matrix factorization, probabilistic approaches
- Problem for model-based methods: how to add data items?
- The first seminar topic is collaborative recommendations

Content-based recommendations (1/2)

- Collaborative recommenders do not need to know anything about the characteristics of the items; content-based recommenders selects on the basis of the **item characteristics**
- Thus: can be used even if there is only one user
- Problem to obtain characterisations of the items
- Content-based recommenders could be viewed as a special case of knowledge-based recommenders (using only item knowledge)
Content-based recommendations

- Typical application domain is recommending text like news articles
- In these cases, the items are “documents”
- Language technology approaches: Viewing documents as vectors of measures (TF-IDF) of keywords
- Example methods: Rocchio’s method, probabilistic methods
- One seminar topic is content-based recommendations

Knowledge-based recommendations

- Conversational systems guide the user to interesting/useful items
  - Constraint-based systems
  - Case-based systems
- One seminar topic is knowledge-based recommendations
Basic Literature


Contacts

- Patrik Floréen: patrik.floreen@cs.helsinki.fi
- Petteri Nurmi: petteri.nurmi@cs.helsinki.fi